Network Intrusion Detection Using Hardware Techniques: A Review

Razan Abdulhammed, Miad Faezipour and Khaled Elleithy
Computer Science and Engineering Department
University of Bridgeport, Bridgeport, CT
rabdulha@my.bridgeport.edu, mfaezipo@bridgeport.edu, elleithy@bridgeport.edu

Abstract

The increasing amount of network throughput and security threat makes intrusion detection a major research problem. In the literature, intrusion detection has been approached by either a hardware or software technique. This work reviews and compares hardware based techniques that are commonly used in intrusion detection systems (IDS) with a special emphasis on modern hardware platforms such as FPGA, GPU, MCP and ASIC.

Introduction

The IDSs can be divided based on the detection approach into two categories: anomaly, and signature based detection. Table 1 shows a comparison between these approaches.

TABLE 1 : A COMPARISON BETWEEN IDS DETECTION APPROACHES

Criteria	Anomaly-Based	Signature-Based
Update	No	Yes
Detection ability	Both Known and unknown attacks	Only Known attacks with very high accuracy
Definition	Use deviation from normal usage pattern to identify intrusions	Use patterns of well-known attacks to identify intrusions
Feature of the system	High False Alarm	Low False Alarm
Implementation requirement	Requires less computation and resources	Requires more computation and resources

The IDS gathers, observes and collects data before the analysis phase using different strategies. These include: host-based and network-based strategies. The Network Intrusion Detection System (NIDS) requires pattern matching, string matching, multimatch packet classification, and a regular expression matching to perform its functions. The utmost reason for shifting from software to hardware is to enable real-time implementation of IDS. A hardware-based intrusion detection system is a scalable method as it is able to inspect packets in high speed networks. We have taken a glance at earlier hardware techniques platforms. Table 2 presents a comparison among hardware platform solutions that include Processor, FPGA, and ASIC. Ternary content addressable memory (TCAM) devices are also widely used in NIDS (Figure 1).

TABLE 2 : COMPARISON AMONG HARDWARE PLATFORM SOLUTIONS

Processors	ASIC	FPGA
Can enhance the aggregate throughput dramatically by using parallel processing Approach	Provide impressively high per-stream throughput	Provide desirable high performance ,flexibility of software and reconfigurable programming
Additional Complexity is introduced in scheduling ,buffering ordering, and load balancing	The applicability is limited by the high implementation cost and low re-programmability	It takes considerable time to resynthesize the design and reprogram the FPGA device

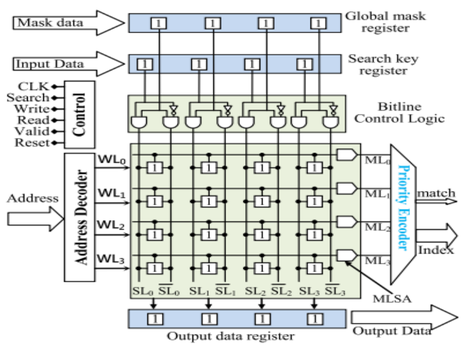

Figure. 1. Typical TCAM device

String and Pattern Matching - Hardware-Based Techniques

The most common method to implement pattern matching in hardware is to use Finite Automata (FA) approaches. Finite Automata (FA) are either Deterministic Finite Automata (DFA) or Non-Deterministic finite automata (NFA). Table 3 presents a sample of the literature. Figure 2 also depicts a sample pattern matching state machine.

TABLE 3 : COMPARISON OF VARIOUS HARDWARE-BASED STRING MATCHING PERFORMANCE SOLUTIONS

Authors	Approach	Platform	Number of Patterns	Pattern Length	Throughput Gbps
Jung	Bit-Split	FPGA	1316	No limit	1.76
Vasiliadies	AC-DFA	GPU	4000	<25	2.3
Lunteren	B_FSM	FPGA	8000	No limit	2.2
Scarpazza	AC-DFA	Cell/B.E	8400	<10	2.5
E Yang	Field Merge	FPGA	6944	<64	4.56
Song	CDFA	ACIS	1785	No limit	6.1
Jiang	Depth-Bounded Pipeline	FPGA	9033	No limit	11.4
Hoang	Leaf-attaching	FPGA	10856	<64	11.8

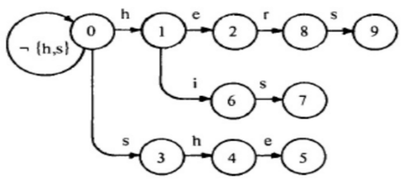

Figure. 2. Pattren matching state machine

Regular Expression Matching- Hardware-Based Techniques

A basic regular expression (Regex) matching engine can be implemented in hardware as a state machine, either a deterministic finite machine or a non-deterministic finite machine. Figure 3 shows the implementation of sample regular expression engine in hardware. Table 4 presents a sample of hardware-based Regex solutions in the literature.

TABLE 4 : COMPARISON OF THE MOST COMMON HARDWARE-BASED REGULAR EXPRESSION MATCHING ENGINE PERFORMANCE SOLUTIONS

Authors	Approach	Non-Meta Chars	Number of LUT per state	Multi-Char Per Cycle	Throughput (Gbps)
Clark et al.	FPGA	17,537	3.1	4	73.8
Sourdis et al	FPGA	69,127	0.66	1	2.42
Yamagaki et al	FPGA	40,896	0.94	4	3.63
Bispo et al.	FPGA	19,580	1.28	1	2.9

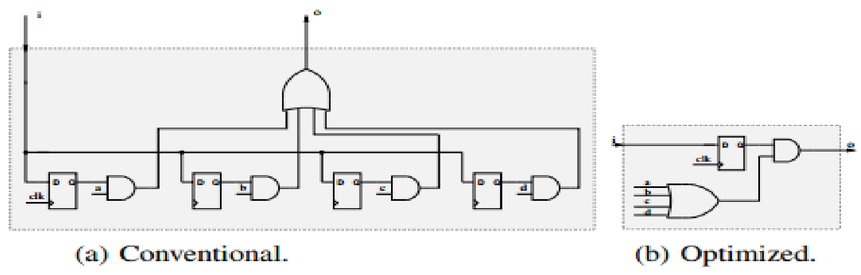

Figure 3. Implementation of “(a|b|c|d)” RegExp.

Packet Classification - Hardware-Based Techniques

Various multi-match packet classification solutions have been introduced. Table 5 shows a comparison of the hardware-based packet classification NIDS techniques. Figure 4 illustrates the main process of packet classification.

TABLE 5 : COMPARISON OF THE PERFORMANCE OF THE STATE OF THE ART HARDWARE SOLUTIONS FOR MULTI-MATCH PACKET CLASSIFICATION IN NIDS

Authors	Algorithm	Platform	Throughput Gbps	Storage	Power mW	SNORT
Song	BV-TCAM	FPGA	10	73.8	17.7	No
Fang	TCAM-SSA	TCAM	20	13	312	No
Faezipour	MX-MN-IP	TCAM	80	13	296	No

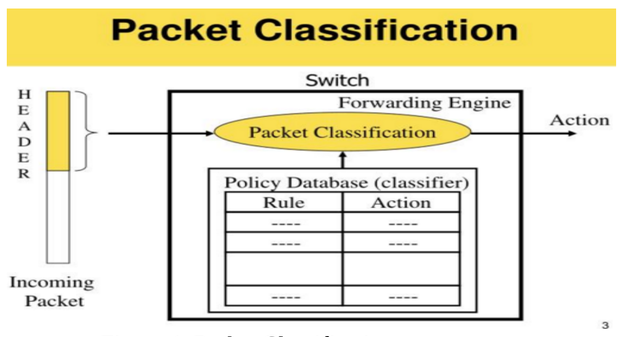

Figure 4. Packet Classification process.

Hardware-Based Intrusion Detection Systems

Table 6 presents a sample of available Hardware-based NIDS in the literature.

TABLE 6 : CLASSIFICATION OF NIDS BASED ON HARDWARE SOLUTION PLATFORMS.

Author	Plat form	Analysis time	Audit material	Audit Features
Pontarelli	FPGA	Real time	Network-Based	UDP Packets,TCP Packets
Quang	FPGA	Real time	Network-Based	IP , Port ,Packets, Octets, Start Time, End Time, Flags
Jaic	FPGA	Real time	Network-Based	Packet payload
Xia	GPU	Offline	Network-Based	TCP/IP packet header features
Kim	GPU	Offline	Network-Based	Packet payloads
Jamshed	GPU	Real time	Host-Based	Packet payloads
Jaehyun	MultiCore Processors	Offline	Network-Based	Packet payloads
Das	FPGA	Real time	Network-Based	Packet payload

Conclusion

The application of hardware for intrusion detection systems has become an active area of research with the support of the modern powerful hardware devices such as FPGA, ASIC, Multicore processor and GPU. However, hardware-based techniques might face real performance challenges due to multiple factors that may affect the implementations of these techniques such as the fiercely rising demands to handle higher traffic rates, analysis, real-time operation, power consumption, memory requirements and the collapse of Moore’s law for sequential processing.

References

1. M. Whitman, and H. Mattord, Principles of information security. 4th Ed, Boston: Cengage Learning. 2011
2. R. Bhatnagar, and U. Shankar, “The proposal of hybrid intrusion detection for defense of sync flood attack in wireless sensor network,” Int. J. Comput.Sci and Eng. Surveys, vol. 3, no 2, pp 31-38, Apr.2012.
3. D. E. Culler, A. Gupta, and J. P. Singh, Parallel Computer Architecture: A Hardware/Software Approach. Morgan Kaufmann Publishers Inc., 1997
4. P. Hunter, “Hardware-based security: FPGA-based devices,” Computer Fraud & Security, vol. 2004, no. 2, pp. 11–12, 2004
5. H. Chen, Y. Chen, and D. H. Summerville, "A Survey on the Application of FPGAs for Network Infrastructure Security," IEEE Commun. Surveys Tutorials, vol.13, no.4, pp.541-561, 2011
6. Yamagaki, N., Sidhu, R., and Kamiya, S. (2008, September). High-speed regular expression matching engine using multi-character NFA. In Field Programmable Logic and Applications, 2008. FPL 2008. International Conference on (pp. 131-136). IEEE.
7. G. Diana, D. Marco, M.P Joao, and B. Koen, “Reconfigurable Computing: Architectures, Tools, and Applications,”: 10th Int. Symp., ARC 2014, Vilamoura, Portugal, April 14-16, 2014.
8. M. Faezipour, M. Nourani, "Wire-Speed TCAM-Based Architectures for Multimatch Packet Classification," IEEE Trans. Comput., vol.58, no.1, pp.5-17, Jan. 2009
9. Jiang, Weirong, and Viktor K. Prasanna. "Field-split parallel architecture for high performance multi-match packet classification using FPGAs."Proceedings of the twenty-first annual symposium on Parallelism in algorithms and architectures. ACM, 2009.
10. Z.K. Baker, and V.K. Prasanna, "Automatic Synthesis of Efficient Intrusion Detection Systems on FPGAs", IEEE Trans.on Dependable and Secure Computing, vol.3, no.4, pp.289-300, Oct.-Dec. 2006.